

ated: Tue, May 5, 1992 11:53 AM EDT Msg: RJC-1710-5579

m: LCARPENTER

MODIS.DATA.TEAM

bj: MODIS SDST Minutes 05/01/92

MODIS Science Data Support Team (SDST) Meeting Minutes 05/01/92

TENDEES: Phil Ardanuy RDC 982-3714  
Lloyd Carpenter RDC 982-3708  
Larry Fishtahler CSC 464-3385  
Al Fleig 900 286-7747  
Tom Goff RDC 982-3704  
Liam Gumley RDC 982-3748  
Janine Harrison 920 286-5324  
Ed Masuoka 920 286-7608  
Frank McGarry 552 286-6846  
Jim Ormsby 974 286-6811  
J-J Pan RDC 982-3738  
Shahin Samadi 920.2/RMS 286-8510  
Lalit Wanchoo STX 513 1682  
Will Webster 920.2 286-4506

XT MEETING: Date Time Building Room  
Friday, May 8 10:00 am 22 G95

PICS:

SOFTWARE DEVELOPMENT IN THE FLIGHT DYNAMICS ENVIRONMENT:

Frank McGarry, GSFC Code 552 presented results and lessons learned over a period of many years in the design, development, and growth of Flight Dynamics software. The emphasis was on the "state of the practice" as opposed to the "state of the art", and software development as opposed to software analysis. The reference was SEL-84-101, Manager's Handbook for Software Development (Revision 1), L. Landis, F. McGarry, S. Waligora, et al., November 1990. Single copies are available from the Systems Development Branch, GSFC Code 552.

Some of the high points of the presentation were:

People are the most important resource. The latest in tools and technology and automation will not compensate for a lack of understanding of the task. Tools can provide a significant benefit when the process is well defined.

Three approaches were studied for detecting errors: Code Reading, Functional Testing, and Structural Testing. Of these three, Code Reading finds the highest percentage of faults, and is the most effective.

The "Cleanroom" approach is to use one group of people for development and a separate one group of people for testing. In a

ly, the error rate dropped and productivity improved when the anroom approach was used. It forces people to use their ught processes (rather than relying on tests to get things ight).

bitrary restrictions on complexity, or lines of code (LOC) per dule, can be counterproductive. Do what makes sense for the ticular case at hand.

ective policies/standards must be written, understood, legacy- ed, enforced and measurable.

acificify the "life cycle" with lists of well defined products and cs to be delivered/accomplished, so there is no doubt about en each phase is completed.

velop formal test plans, and provide for an independent testing m.

ective management/control depends upon availability and use of trics including resource expenditures (people, computer time), ors, changes, software growth, system size estimates, etc. gular updates (weekly or monthly) are necessary for identifying blems at an early stage.

breakdown of total effort in software development was found e:

Preliminary Design	15%
Detailed Design	17%
Implementation	26%
System Testing	23%
Acceptance Testing	19%

he Flight Dynamics area, the cost of reusing code is 20% of cost of developing new code (in Fortran).

y have experience in porting code between DEC and IBM chines. Building in machine independence has not been cessful in their experience.

ODING RECOMMENDATIONS: J. J. Pan presented a draft of Liam mley's RADIANCE program as a proposed example of a Fortran gram to be included in the coding recommendations for MODIS ence Team Members. Four of the modules are given together h the FTNCHECK output for Fortran 77 standard and portability cking.

OTHER: Action Items, and reports on MAS, Cadre and NetCDF e included in the handout, but were not discussed due to lack ime. Copies of the draft version of the Team Leader's Science mputing Facility Plan were also distributed.

#### TION ITEMS:

24/92 [Lloyd Carpenter] Prepare the Team Leader's Software  
Data Management Plan for review. STATUS: Open. Due Date:  
May 10, 1992.

24/92 [Lloyd Carpenter] Prepare the Team Leader's Science  
Computing Facility Plan for review. (Copies of the current draft  
version were provided along with the handout.) STATUS: Open.  
Due Date: May 10, 1992.

24/92 [Tom Goff] Develop a detailed schedule through to the  
delivery of Version 1 to the DAAC for Level-1A and -1B software  
design and development, identification of risk areas in Level-1A  
and -1B design, and prototyping of risks. STATUS: Open. Due  
Date:

24/92 [J. J. Pan] Develop a detailed schedule for the Level-2  
Processing Shell design and development, identification of risk  
areas in the Level-2 Processing Shell design and development, and  
prototyping of risks, through to the delivery of Version 1 to the  
DAAC. STATUS: Open. Due Date:

24/92 [J. J. Pan] Develop a detailed schedule for a typical  
algorithm integration into the Level-2 processing shell. STATUS:  
Open. Due Date:

24/92 [Lloyd Carpenter & Team] Develop a staffing plan for  
accomplishment of the tasks shown on the schedule. STATUS:  
Open. Due Date: